

Hi-C Data User Guide

0.	Setup	3
	a) Directory Structure and IDL Tree	3
	b) Batch Processing	3
	c) Updates to Version 3	3
4k x 4k SET:		
1.	Convert Level 0 to Level 0.5	4
	a) Purpose	4
	b) Program	4
	c) Example	4
	d) Output	4
	e) Mapping	5
2.	Convert Level 0.5 to Level 1.0	5
	a) Purpose	5
	b) Program	5
	c) Example	5
	d) Output	6
	e) Mapping	6
3.	Further Prepping Level 1.0	6
	a) Purpose	6
	b) Program	6
	c) Example	6
	d) Output	7
	e) Mapping	7
4.	Convert Level 1.0 to Level 1.5	7
	a) Purpose	7
	b) Program	7
	c) Example	8
	d) Output	8
	e) Mapping	8
5.	Further Prepping Level 1.5	9
	a) Purpose	9
	b) Program	9
	c) Example	9
	d) Output	9
	e) Mapping	9
6.	Summary of VSO Availability	10
	a) Graphic Summary of Basic Level Outputs	10
1k x 1k SET:		
7.	Convert Level 0 to Level 0.5 (1K)	11
	a) Purpose	11
	b) Program	11

c)	Example	11
d)	Output	12
e)	Mapping	12
8.	Convert Level 0.5 to Level 1.0 (1K)	12
a)	Purpose	12
b)	Program	13
c)	Example	13
d)	Output	13
e)	Mapping	13
9.	Further Prepping Level 1.0 (1K)	14
a)	Purpose	14
b)	Program	14
c)	Example	14
d)	Output	14
e)	Mapping	14
10.	Convert Level 1.0 to Level 1.5 (1K)	15
a)	Purpose	15
b)	Program	15
c)	Example	15
d)	Output	16
e)	Mapping	16
11.	Further Prepping Level 1.5 (1K)	16
a)	Purpose	16
b)	Program	17
c)	Example	17
d)	Output	17
e)	Mapping	17
12.	Further Prepping Level 2.0 (1K)	17
a)	Purpose	17
13.	Summary of VSO Availability (1K)	18
a)	Graphic Summary of Basic Level Outputs	18
14.	Reference Masks	19
a)	Purpose	19
b)	Create a Dust Mask	19
c)	Create a Quadrant Mask	19
15.	Additional Notes	20
a)	THROUPT	20

RED TEXT INDICATES CHANGES FROM PREVIOUS VERSION.

0. Setup

a) Directory Structure and IDL Tree

In order to run the programs listed within this document, you will need to include all of Hi-C IDL programs under your IDL tree. In other words, you will need to perform an `ssw_upgrade` and include "HIC" as an instrument variable in your particular shell login script (or use `ssw_instr`).

*** Testing statements throughout the document refer to the files and directory structure made available through the MSFC Sol1 Hi-C directory. ***

Proper directory structure for the following testing statements:

```
Hi-C >
  Level0
  Level0.5
  Level1.0
  Level1.5
  1K >
    Level0
    Level0.5
    Level1.0
    Level1.5
```

b) Batch Processing

The full sets can be processed by running a single IDL program; however, it can currently only be fully run at NASA/MSFC on the data server.

```
IDL>
  process_hic[, dir=dir, /k1, /k4]
```

c) Updates to Version 3

- The time stamp was originally corrected based on the `.idx` time stamp. This was determined to be an incorrect time stamp, and should have instead been read from the original FITS headers. This was corrected in versions 2 and 3 of the Hi-C data.

- Two center rows were found to be missing from the output of the CCD. Version 3 of the data has this issue corrected. The center coordinates had to be adjusted based on the addition of these rows. The coordinates also had to be adjusted due to the time stamp fix since a different Hi-C image had to be used as the reference frame.

- Fine alignment shifts were applied.

- Normalization was applied to scale the 1K and 4K sets together. The stacking method of the 1K set was changed to averaging and. An additional dust map was also created.

- “Movie-quality” sets were created -- Level 2.0 & 2.0_stacked. These were developed by matching the detector quadrants in the high-noise 1K set and zeroing out the negative noise.

- The Level 1.5 & 2.0 sets are aligned with tracking to remove the drift from the set of images. The center coordinates still reflect the true pointing. (*Only the X-coordinate is adjusted for drift due to the small roll angle and short observing period.*)

1. Convert Level 0 to Level 0.5

a) Purpose

To fix the information contained within the headers of the FITS and index files and transpose the data delivered straight from the rocket. The method for deriving the corrected information is described within HIC_4k_Level0.5_Development.pdf. Note that this step is performed at NASA/MSFC as an intermediary step to the calibrated Level 1.0 set. It can be made available upon request.

b) Program

Run hic_level05.pro on Level 0 data to convert to Level 0.5.

c) Example

```
>> Sol1/Hi-C/Level0/
```

```
IDL>
```

```
;;... dir = '/volumes/Sol1/Hi-C/'
cd, dir + 'Level0/'
files0 = file_search('*.*fit')
```

;- Index files (.idx), which must be in the same directory as the FITS, are located automatically.

```
hic_level05, files0, savedir='../Level0.5'
```

d) Output

- Level 0.5 FITS files
- Naming scheme: HiC_L0.5_v3_[time stamp].fit

The Level 0.5 FITS data are already available in Sol1/Hi-C/Level0.5/. These files contain raw images with correct header information.

e) Mapping

```
>> Sol1/Hi-C/Level0.5/
```

```
IDL>
  cd, '../Level0.5/'
  files05 = file_search('*L0.5*')
  mreadfits, files05, hindex05, hdata05 ;[, head=head05, /quiet]

  ; - head05 will be the last file's FITS header
  index2map_ss, hindex05, hdata05, hic_l05_map
  save, filename='hic_l05_map_v3.sav', hic_l05_map

  ;*** Or restore the map that is already available in Sol1/Hi-C/maps/ ***
  restore, 'hic_l05_map_v3.sav', /v
  my_stepper, hic_l05_map, /k4 ;[ or ... /over, gam=.4, res=.25]
```

2. Convert Level 0.5 to Level 1.0

a) Purpose

To provide a data set that is dark-subtracted, flat-fielded, **dust-hidden**, and appropriately cropped. The data can be optionally normalized, and/or unrolled. *This calibrated set is made available to the community via the VSO.*

**** Note that it is not recommended to go from Level 0 to Level 1.0 due to incorrect and missing header information. ****

b) Program

Run `hic_prep.pro` on Level 0.5 data to convert to Level 1.0.

c) Example

```
>> Sol1/Hi-C/Level0.5/
```

```
IDL>
  cd, '../Level0.5/'
  files05 = file_search('*L0.5*')
  hic_level1, files05, savedir='../Level1.0'
```

d) Output

- Level 1.0 FITS files
- Naming scheme: HiC_L1.0_v3_[time stamp].fit

The basic Level 1.0 FITS data are already available in Sol1/Hi-C/**Level1.0/**. These images have been dark-subtracted, flat-fielded, **dust-hidden**, and appropriately cropped -- **not** normalized, unrolled, or aligned.

e) Mapping

>> Sol1/Hi-C/**Level1.0/**

```
IDL>
  cd, '../Level1.0/'
  files1 = file_search('*L1.0*')
  mreadfits, files1, hindex1, hdata1 ;[, head=head1, /quiet]
  index2map_ss, hindex1, hdata1, hic_l1_map
  save, filename='hic_l1_map_v3.sav', hic_l1_map

  ;*** Or restore the map that is already available in Sol1/Hi-C/maps/ ***
  ;restore, 'hic_l1_map_v3.sav', /v
  my_stepper, hic_l1_map, /k4 ;[ or ... /over, gam=.4, res=.25]
```

3. Further Prepping Level 1.0 [OPTIONAL]

a) Purpose

To normalize, ~~hide dust from~~, and/or unroll a Level 1.0 data set which was previously dark-subtracted, flat-fielded, dust-hidden, and cropped (at least).

*** Note that previously-unrolled data cannot be dust-hidden. ***

b) Program

Run hic_prep.pro on Level 1.0 data to further prep.

c) Example

>> Sol1/Hi-C/**Level1.0/**

```
IDL>
  cd, '../Level1.0/'
```

```

files1 = file_search('*L1.0*')
hic_prep, files1, -1, hindex1, hdata1, /savefits ;[, /normalize,
/unroll]

```

d) Output

- Level 1.0 FITS files
- Naming scheme: HiC_L1.0_v3_[time stamp].fit

Level 1.0 FITS data that have been dark-subtracted, flat-fielded, dust-hidden, and appropriately cropped and optionally normalized, and/or unrolled.

e) Mapping

```
>> Sol1/Hi-C/Level1.0/
```

```

IDL>
  cd, '../Level1.0/'
  files1 = file_search('*L1.0*')
  mreadfits, files1, hindex1, hdata1 ;[, head=head1, /quiet]
  index2map_ss, hindex1, hdata1, hic_l1_map
  ;;...save, filename='hic_l1_map.sav', hic_l1_map
  my_stepper, hic_l1_map, /k4 ;[ or ... /over, gam=.4, res=.25]

```

4. Convert Level 0.5 or Level 1.0 to Level 1.5

a) Purpose

To provide a data set that is dark-subtracted, flat-fielded, appropriately cropped, dust-hidden, **normalized**, co-aligned **with tracking**, and **absorption-compensated**. The data can be optionally unrolled (**not recommended due to image degradation**).

**** These images are very large. Aligning the set is not a fast process and is computationally expensive. It is recommended that this set be downloaded via VSO (or accessed on Sol1). ****

b) Program

Run `hic_level15.pro` on Level 0.5 or Level 1.0 data to convert to Level 1.5.

**** Do not run on un-dusted, unrolled data because `hic_prep` cannot hide the dust from unrolled data. ****

c) Example

```
>> Sol1/Hi-C/Level1.0/
```

```
IDL> cd, '../Level1.0/'
      files1 = file_search('*L1.0*')
      hic_level15, files1, hindex15, hdata15, savedir='../Level1.5' ;[,
/normalize, /unroll]
```

***** Or *** (less recommended)**

```
>> Sol1/Hi-C/Level0.5/
```

```
IDL> cd, '../Level0.5/'
      files05 = file_search('*L0.5*')
      hic_level15, files05, hindex15, hdata15, savedir='../Level1.5' ;[,
/normalize, /unroll]
```

**** Note that previously-unrolled data cannot be dust-hidden and the /fast option for hiding dust is not accepted (-- it will be re-dusted the full way in that case). ****

d) Output

- Level 1.5 FITS files
- Naming scheme: HiC_L1.5_v3_[time stamp].fit

The basic Level 1.5 FITS data is already available in Sol1/Hi-C/Level1.5/. These images have been dark-subtracted, flat-fielded, appropriately cropped, dust-hidden, **normalized**, and co-aligned **with tracking** -- **not** unrolled.

e) Mapping

```
>> Sol1/Hi-C/Level1.5/
```

```
IDL> cd, '../Level1.5/'
      files15 = file_search('*L1.5*')
      mreadfits, files15, hindex15, hdata15 ;[, head=head15, /quiet]
      index2map_ss, hindex15, hdata15, hic_l15_map
      save, filename='hic_l15_map_v3.sav', hic_l15_map
```

```
;*** Or restore the map that is already available in Sol1/Hi-C/maps/ ***
restore, 'hic_l15_map_v3.sav', /v
my_stepper, hic_l15_map, /k4 ;[ or ... /over, gam=.4, res=.25]
```

5. OPTIONAL Further Prepping Level 1.5

a) Purpose

To normalize and/or unroll a Level 1.5 data set which was previously dark-subtracted, flat-fielded, cropped, dust-hidden, and aligned with tracking (at least).

b) Program

Run `hic_prep.pro` on Level 1.5 data to further prep.

c) Example

```
>> Sol1/Hi-C/Level1.5/
IDL>
  cd, '../Level1.5/'
  files15 = file_search('*L1.5*')
  hic_prep, files15, -1, hindex15, hdata15, /savefits ;[, /unroll]
```

d) Output

- Level 1.5 FITS files
- Naming scheme: `HiC_L1.5_v3_[time stamp].fit`

Level 1.5 FITS data that have been dark-subtracted, flat-fielded, cropped, dust-hidden, normalized, and co-aligned and/or unrolled.

e) Mapping

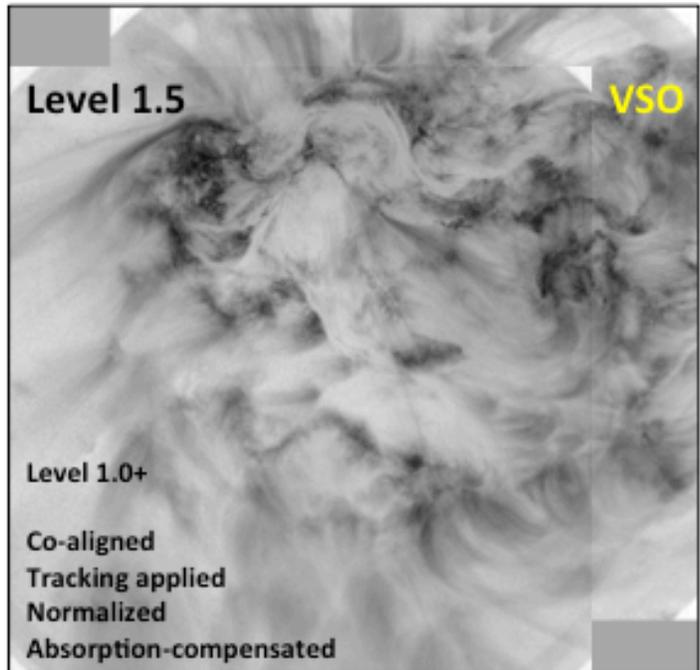
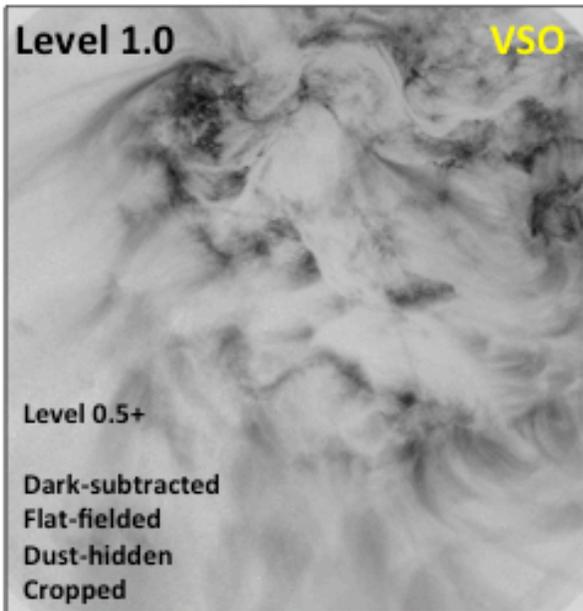
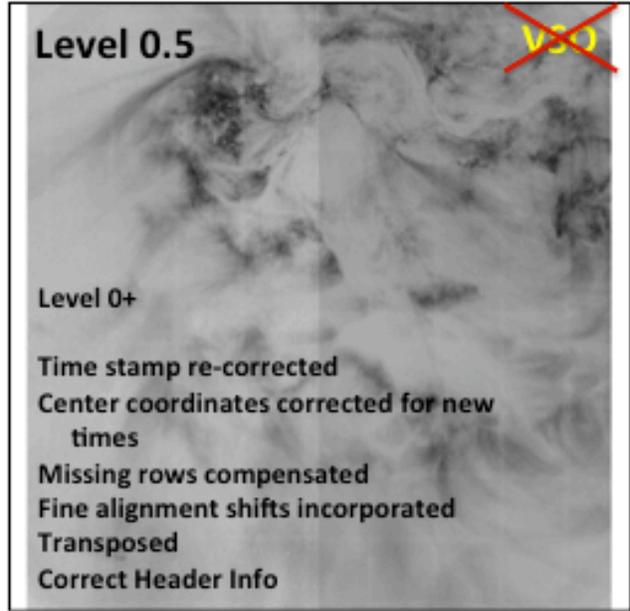
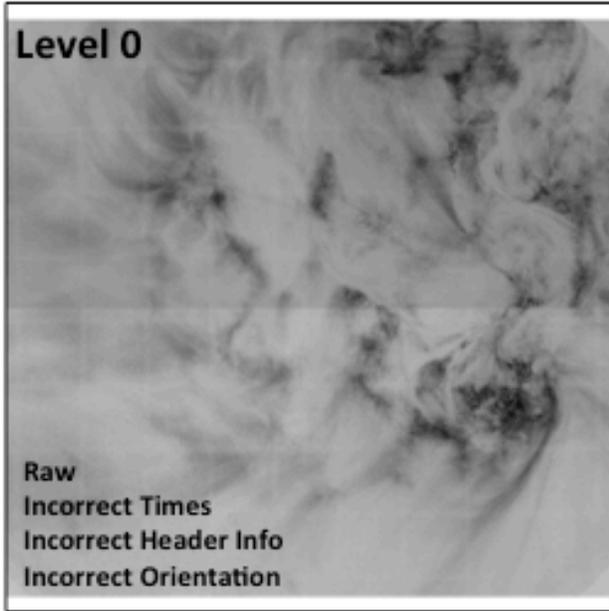
```
>> Sol1/Hi-C/Level1.5/
IDL>
  cd, '../Level1.5/'
  files15 = file_search('*L1.5*')
  mreadfits, files15, hindex15, hdata15 ;[, head=head15, /quiet]
  index2map_ss, hindex15, hdata15, hic_l15_map
```

```
;;...save, filename='hic_l15_map.sav', hic_l15_map
my_stepper, hic_l15_map, /k4 ;[ or .../over, gam=.4, res=.25]
```

6. Summary of VSO Availability

The basic versions of Levels 1.0 and 1.5 are available to the community via VSO.

a) Graphic Summary of Basic Level Outputs



7. Convert Level 0 to Level 0.5 (1K)

a) Purpose

To fix the information contained within the headers of the FITS and index files and transpose the data delivered straight from the rocket. The pointings were determined by comparing to the pre-determined 4K data in Section 1 (Level 0.5). The offsets between the first 1K image and the last 4K image (post-repointing) are:

$$\begin{aligned} 1k.xc &= 1k.xc + 12.81 \text{ [arcsec]} \\ 1k.yc &= 1k.yc - 2.9516 \text{ [arcsec]} \end{aligned}$$

There are currently no relative offsets derived via correlation methods for the 1K set; therefore, none are applied. Any residual jitter in the pointing is not significant.

**** Note that the last image (#148) has been removed from the set due to extreme atmospheric absorption and poor pointing information. ****

To obtain a set from the earlier portion of the flight, the same sub-region is extracted from the Level 0.5 4K set and combined with the 1K-only set.

Note that this step is performed at NASA/MSFC as an intermediary step to the calibrated Level 1.0 set. It can be made available upon request.

b) Program

Run `hic_level05.pro` on Level 0 data to convert *1K-only set* to Level 0.5.

Run `hic_ext1k.pro` to extract the 1K region from the Level 0.5 4K set.

**** Note that the programs must be run in this order because one of the Level 0.5 1K FITS images is required to properly extract from the Level 0.5 4K set. ****

c) Example

```
>> Sol1/Hi-C/1K/Level0/
```

```
IDL>
```

```
;;...dir1k = '/Volumes/sol1/Hi-C/1K/'
```

```
cd, dir1k + 'Level0/'
```

```
files1k0 = file_search('*.fit')
```

; - Index files (.idx), which must be in the same directory as the FITS, are located automatically.

```
hic_level05, files1k0, savedir='../Level0.5'
```

```

files1k05 = file_search('../Level0.5/*L1k0.5*')
files4k05 = file_search('../../Level0.5/*L0.5*')
hic_ext1k, files4k05, files1k05[0], savedir='../Level0.5'

```

d) Output

- Level 0.5 1K FITS files: extractions from the earlier 4K set + 1K-only set
- Naming scheme: HiC_L1k0.5_[time stamp].fit

The Level 0.5 1K FITS data are already available in Sol1/Hi-C/**1K/Level0.5/**. These files contain raw images with correct header information.

e) Mapping

>> Sol1/Hi-C/**1K/Level0.5/**

```

IDL>
  cd, '../Level0.5/'
  files1k05 = file_search('*L1k0.5*')
  mreadfits, files1k05, hindex1k05, hdata1k05 ;[, head=head1k05, /quiet]
  ; - head1k05 will be the last file's FITS header
  index2map_ss, hindex1k05, hdata1k05, hic_l1k05_map
  save, filename='hic_l1k05_map_v3.sav', hic_l1k05_map
  ;*** Or restore the map that is already available in Sol1/Hi-C/maps/ ***
  restore, 'hic_l1k05_map_v3.sav', /v
  my_stepper, hic_l1k05_map, /k1 ;[ or ... /over, gam=.4, res=.5]

```

8. Convert Level 0.5 to Level 1.0 (1K)

a) Purpose

To provide a data set that is dark-subtracted, flat-fielded, **and dust-hidden**. The data can be optionally normalized and/or unrolled.

To obtain a set from the earlier portion of the flight, the same sub-region is extracted from the Level 1.0 4K set and combined with the 1K-only set. **The calibrated Level 1.0 set is available via the VSO.**

***** Note that it is not recommended to go from Level 0 to Level 1.0 due to incorrect and missing header information. *****

b) Program

Run `hic_ext1k.pro` to extract the 1K region from the Level 1.0 4K set, which has already been initially prepped (with a 4K-specific dark frame).

Run `hic_prep.pro` on the *1K-only* Level 0.5 data to convert to Level 1.0.

c) Example

```
>> Sol1/Hi-C/1K/Level0.5/
```

```
IDL> cd, '../Level0.5/'
      files1k05 = file_search('*L1k0.5*')
      files4k1 = file_search('../../Level1.0/*L1.0*')
      hic_level1, files1k05, files4k=files4k1, savedir='../Level1.0'
```

d) Output

- Level 1.0 1K FITS files: extractions from the earlier 4K set + 1K-only set
- Naming scheme: HiC_L1k1.0_[time stamp].fit

The basic Level 1.0 1K FITS data are already available in `Sol1/Hi-C/1K/Level1.0/`. These images have been dark-subtracted, flat-fielded, **dust-hidden** -- **not** normalized or unrolled.

e) Mapping

```
>> Sol1/Hi-C/1K/Level1.0/
```

```
IDL> cd, '../Level1.0/'
      files1k1 = file_search('*L1k1.0*')
      mreadfits, files1k1, hindex1k1, hdata1k1 ;[, head=head1k1, /quiet]
      index2map_ss, hindex1k1, hdata1k1, hic_l1k1_map
      save, filename='hic_l1k1_map_v3.sav', hic_l1k1_map
      ;*** Or restore the map that is already available in Sol1/Hi-C/maps/ ***
      restore, 'hic_l1k1_map_v3.sav', /v
      my_stepper, hic_l1k1_map, /k1 ;[ or ... /over, gam=.4, res=.5]
```

9. OPTIONAL Further Prepping Level 1.0 (1K)

a) Purpose

To normalize, ~~hide dust from,~~ and/or unroll a Level 1.0 1K data set which was previously dark-subtracted and flat-fielded (at least).

**** Note that previously-unrolled data cannot be dust-hidden. ****

b) Program

Run hic_prep.pro on Level 1.0 data to further prep.

c) Example

```
>> Sol1/Hi-C/1K/Level1.0/
```

```
IDL> cd, '../Level1.0/'
      files1k1 = file_search('*L1k1.0*')
      hic_prep, files1k1, -1, hindex1k1, hdata1k1, /savefits ;[, /normalize,
/unroll]
```

d) Output

- Level 1.0 1K FITS files: assuming extractions from the earlier 4K set + 1K-only set
- Naming scheme: HiC_L1k1.0_[time stamp].fit

Level 1.0 1K FITS data that have been dark-subtracted, flat-fielded, and dust-hidden and optionally normalized and/or unrolled.

e) Mapping

```
>> Sol1/Hi-C/1K/Level1.0/
```

```
IDL> cd, '../Level1.0/'
      files1k1 = file_search('*L1k1.0*')
      mreadfits, files1k1, hindex1k1, hdata1k1 ;[, head=head1k1, /quiet]
      index2map_ss, hindex1k1, hdata1k1, hic_l1k1_map
      ;;... save, filename='hic_l1k1_map.sav', hic_l1k1_map
      my_stepper, hic_l1k1_map, /k1 ;[ or ... /over, gam=.4, res=.5]
```

10. Convert Level 0.5 or Level 1.0 to Level 1.5 (1K)

a) Purpose

To provide a data set that is dark-subtracted, flat-fielded, dust-hidden, **normalized, absorption-compensated**, co-aligned **with tracking**, and combined with the Level 1.5 4K extractions. The data can be optionally unrolled (**not recommended due to image degradation**). **The processed Level 1.5 is available via VSO.**

b) Program

Run `hic_ext1k.pro` to extract the 1K region from the Level 1.5 4K set. *****Note that this step is necessarily done before `hic_level15.pro`.*****

Run `hic_level15.pro` on Level 0.5 or Level 1.0 1K data to convert to Level 1.5.

***** Do not run on un-dusted, unrolled data because `hic_prep` cannot hide the dust from unrolled data. *****

c) Example

>> Sol1/Hi-C/1K/Level1.0/

```
IDL>
  cd, '../Level1.0/'
  files1k1 = file_search('*L1k1.0*')
  files4k15 = file_search('../../Level1.5/*L1.5*')
  hic_ext1k, files4k15, files1k1[0], savedir='../Level1.5'
  extfiles4k15 = file_search('../Level1.5/*L1k1.5*')
  hic_level15, files1k1, hindex1k15, hdata1k15, extfiles4k=extfiles4k15,
  savedir='../Level1.5' ;[/unroll]
```

***** Or *** (less recommended)**

>> Sol1/Hi-C/1K/Level0.5/

```
IDL>
  cd, '../Level0.5/'
  files1k05 = file_search('*L1k0.5*')
  files4k15 = file_search('../../Level1.5/*L1.5*')
  hic_ext1k, files4k15, files1k05[0], savedir='../Level1.5' =
```

```

extfiles4k15 = file_search('../Level1.5/*L1k1.5*')
hic_level15, files1k05, hindex1k15, hdata1k15,
extfiles4k=extfiles4k15, savedir='../Level1.5' ;[/unroll]

```

*** Note that previously-unrolled data cannot be dust-hidden and the /fast option for hiding dust is not accepted (-- it will be re-dusted the full way in that case). ***

*** Note that the Level 1.5 4K extractions are over-written in this example after the full set is aligned. ***

d) Output

- Level 1.5 1K FITS files: extractions from the earlier 4K set + 1K-only set
- Naming scheme: HiC_L1k1.5_[time stamp].fit

The basic Level 1.5 1K FITS data are already available in Sol1/Hi-C/**1K/Level1.5/**. These images have been dark-subtracted, flat-fielded, dust-hidden, **normalized**, co-aligned **with tracking, and absorption-compensated** -- **not** unrolled.

This set is not stacked.

e) Mapping

```
>> Sol1/Hi-C/1K/Level1.5/
```

```

IDL>
  cd, '../Level1.5/'
  files1k15 = file_search('*L1k1.5*')
  mreadfits, files1k15, hindex1k15, hdata1k15 ;[, head=head1k15, /quiet]
  index2map_ss, hindex1k15, hdata1k15, hic_l1k15_map
  save, filename='hic_l1k15_map_v3.sav', hic_l1k15_map
  ;*** Or restore the map that is already available in Sol1/Hi-C/maps/ ***
  restore, 'hic_l1k15_map_v3.sav', /v
  my_stepper, hic_l1k15_map, /k1 ;[ or ... /over, gam=.4, res=.5

```

11. OPTIONAL Prepping Level 1.5 (1K)

a) Purpose

To unroll a Level 1.5 data set which was previously dark-subtracted, flat-fielded, dust-hidden, normalized, absorption-compensated, and aligned with tracking (at least).

b) Program

Run `hic_prep.pro` on Level 1.5 data to further prep.

c) Example

```
>> Sol1/Hi-C/1K/Level1.5/
```

```
IDL> cd, '../Level1.5/'
      files1k15 = file_search('*L1k1.5*')
      hic_prep, files1k15, -1, hindex1k15, hdata1k15, /savefits ;[, /unroll]
```

d) Output

- Level 1.5 1K FITS files: assuming extractions from the earlier 4K set + 1K-only set
- Naming scheme: `HiC_L1k1.5_[time stamp].fit`

Level 1.5 1K FITS data that have been dark-subtracted, flat-fielded, dust-hidden, normalized, absorption-compensated, and co-aligned with tracking and optionally unrolled.

e) Mapping

```
>> Sol1/Hi-C/1K/Level1.5/
```

```
IDL> cd, '../Level1.5/'
      files1k15 = file_search('*L1k1.5*')
      mreadfits, files1k15, hindex1k15, hdata1k15 ;[, head=head1k15, /quiet]
      index2map_ss, hindex1k15, hdata1k15, hic_l1k15_map
      ;;...save, filename='hic_l1k15_map.sav', hic_l1k15_map
      my_stepper, hic_l1k15_map, /k1 ;[ or ... /over, gam=.4, res=.5]
```

12. Convert Level 0.5 or Level 1.0 to Level 1.5_stacked (1K)**a) Purpose**

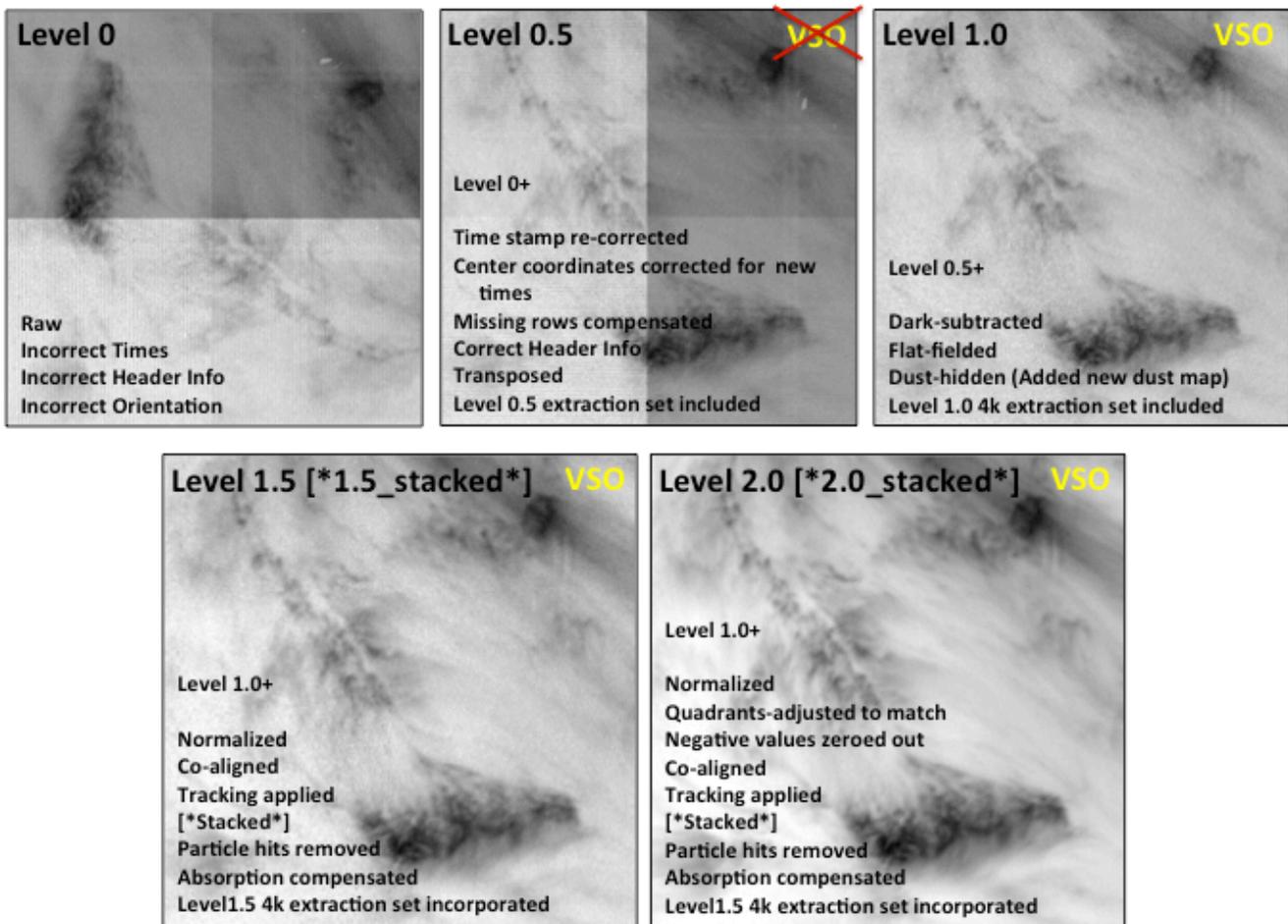
To provide a data set that is dark-subtracted, flat-fielded, dust-hidden, normalized, absorption-compensated, co-aligned with tracking, stacked, and combined with the 4K extractions. The data can be optionally unrolled (not recommended). The processed Level 1.5_stacked is available via VSO.

The method for converting to the Level 1.5_stacked set is the same as that for converting to Level 1.5 in the previous section except to add the keyword /stack to the hic_level15 program call. The 1k set is stacked by co-adding and normalizing (i.e., averaging).

13. Summary of Data Availability (1K)

The basic versions of the Levels 1.0, 1.5, and 1.5_stacked 1K data are available to the community via VSO. Additionally available Level 2.0 and 2.0_stacked sets have been made available and are considered “movie-quality”. The pixel values, however, are not convertible to DN/s.

a) Graphic Summary of Basic Level Outputs



14. Reference Masks

a) Purpose

To create reference images that indicate how pixels were changed via hiding the dust or raising the quadrant level intensities.

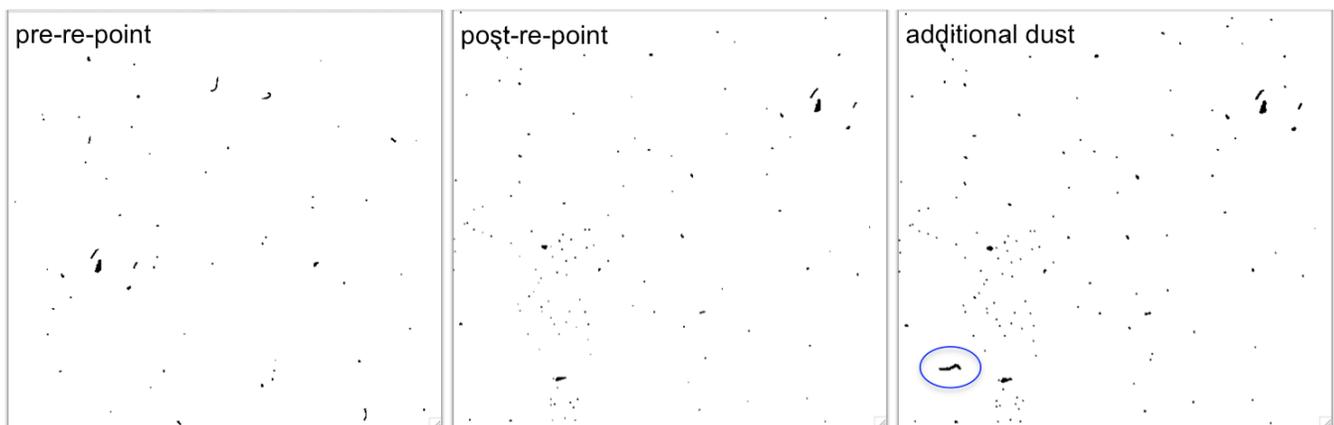
b) Create a Dust Mask

Some levels of the data have an associated dustinfo file. The following describes how to use the dustinfo file to create a reference mask.

This example will apply to the Level 1.5 1K set and assumes that you have the map available. If the map is not available, you can create one from the FITS files (see section 10.e).

```
IDL>
  cd, '[~/Hi-C/1K/Level1.5/'
  restore, 'hic_1k15_map_v3.sav', /v
  restore, 'dustinfo_1k15.sav', /v
  frame = 40 ;as an example
  dustmask = hic_get_dustmask(dustinfo, hic_1k15_map[frame], /sametime)
  my_window, dustmask, /k1
```

The hic_map file does not have to correspond to the same level as the dustinfo file and can even be a 4K map, but it will only show the dust in the 1K region if using a 1K dustinfo file.



c) Create a Quadrant Mask

Some levels of the data have an associated quadinfo file. The following describes how to use the quadinfo file to create a reference mask.

This example will apply to the Level 1.5 1K set and assumes that you have the map available. If the map is not available, you can create one from the FITS files (see section 10.e).

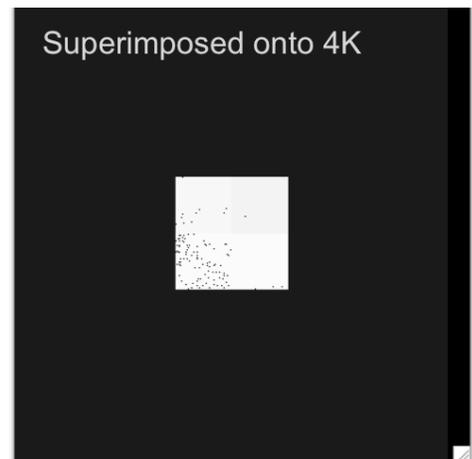
```
IDL>
  cd, '[~/Hi-C/1K/Level1.5/'
  restore, 'hic_1k15_map_v3.sav', /v
  restore, 'quadinfo_1k15.sav', /v
  frame = 40 ;as an example
  quadmask = hic_get_quadmask(quadinfo, hic_1k15_map[frame], /sametime)
  my_window, quadmask, /k1
```

Because the quadrant adjustments were not applied to the 4K data set, the /sametime keyword is not applicable to frames 0 – 35 of any of the full 1K sets that include the 4K extractions.

The black dots indicate the negative pixels that were set to 0.

The shading indicates the scaling level of the quadrant.

Different mask per Hi-C image.



15. Additional Note(s)

a) THROUPUT

The THROUPUT FITS header keyword is archival information and not applicable to the current data sets.